These are the proceedings of the workshop on Producing High Performance and Sustainable Software for Molecular Simulation held at SC15: The International Conference for High Performance Computing, Networking, Storage and Analysis on November 20th, 2015 in Austin, Texas. It was organised by the APES project, a jointly US (NSF CHE-1363320) and UK (EPSRC EP/K040138/) funded collaboration that developed and deployed robust and sustainable software for advanced potential energy surfaces.

The aim of the workshop was to explore the challenges associated with improving the performance and sustainability of molecular simulation software and thereby enabling the continued delivery of cutting-edge science on constantly evolving hardware platforms. It aimed to bring together developers and users of molecular simulation software, experts in high-performance computing, and researchers of relevant numerical methods and algorithms.

The workshop provided a forum for experts to share insights and strategies aimed at continuing to produce software that can push the boundaries in terms of timescales, system sizes and accuracy and to discuss their relative merits and tradeoffs, particularly in light of hardware trends and with an eye on software sustainability. It functioned as an opportunity for HPC experts to better understand the challenges faced in developing software for use by a large group of users of HPC resources within both academia and industry, whilst users gained a better understanding of the state-of-the-art performance characteristics of different packages and how to use these and the machines they run on most efficiently in their research. The workshop also encouraged the exchange of views of users and developers of molecular simulation software with those of other types of high-performance scientific software facing similar challenges.

At the same time as considering how to enable high performance, participants - both developers and users - were provided with the opportunity to reflect on and discuss the impact that software development practices and different ways of meeting the above challenges have on the maintainability, portability, extensibility, interoperability and usability of codes over the longer term.

The workshop focused on the following topics:

- Current and future performance bottlenecks
- Efficient use of large-scale and heterogeneous compute resources including accelerators
- New or improved (parallel) numerical methods and algorithms
- Parallelization strategies and load balancing
- Benchmarking and performance comparisons
- Libraries and APIs
- Code extensibility and integration of new functionality (within a package and across packages)
- Interoperability of software packages
- Portability of code and of code performance
- Software development practices and collaborative models

- Testing
- Workflows

Contributions were sought documenting state of the art developments, current bottlenecks, anticipated challenges, and priorities for future research and funding for molecular simulation software. The following kinds of contributions were sought:

- *Hot topics*: recent work with (potential for) high impact on performance and / or sustainability of molecular simulation software (e.g. algorithms, forcefield models, parallelisation schemes and use of programming models, software development practices, exploiting new hardware).

- *Case studies*: perspectives on achieving performance and / or sustainability based on experience in particular application areas, software development initiatives, etc. (e.g. interplay between optimisation and portability, analysis of goals and challenges, routes and barriers to success - technical, funding).

The workshop featured three experienced invited speakers who agreed to share their insights and vision on the topics of the workshop (their presentations are included in the proceedings):

Erik Lindahl
GROMACS, KTH Royal Institute of Technology and Stockholm University

Molecular Simulation in the Exascale Era:
Acceleration, Task Parallelism & Ensemble Simulations

Molecular dynamics simulations have evolved from a niche theoretical method into a standard technique widely used everywhere from physics and materials science to chemistry and life science. While a large part of the revolution is due to faster hardware, it is arguably the emergence of high performance standardized simulation software codes that has made the technique so remarkably successful in applications.

Compared to only a decade or two ago, simulation codes have improved tremendously and many of the packages scale to very large number of processors. However, the field is also facing some acute challenges. Most application problems in biomolecular simulation are small, and the bottleneck is usually that we need to cover longer timescales. This means weak scaling will not help much. Molecular dynamics can have an iteration time for a time step under 1ms, which makes strong scaling exceptionally difficult as each generation of hardware has more cores - but they tend to run at lower clock frequency.

Over the last decade we have redesigned the entire parallelization engine of the GROMACS molecular simulation toolkit to achieve not only strong scaling, but high absolute efficiency for all types of hardware. The code employs a minimal-communication neutral territory domain decomposition combined with dual-layer MPI & OpenMP parallelization. GROMACS has long relied on explicit SIMD (single-instruction, multiple-data) parallelism for

performance, and in the latest version we have developed an entirely new algorithm to evaluate nonbonded interactions (replacing the classical neighborlist) to achieve portability between traditional CPU units, SIMD code, and CUDA as well as OpenCL accelerators - including a new portable SIMD layer covering every single architecture on top-500.

This multi-level parallelism performs very well, but there are a number of very large challenges remaining for the entire field that we have only begun scratching: To achieve an order-of-magnitude speedup on fixed-size problems we need to reach iteration frequencies down to 100 microseconds, which will require entirely new heterogeneous acceleration approaches. We need algorithms for very fine-grained task parallelism and implementations that work with sub-microsecond latencies for each task, we need algorithms that can make full use of increasingly wide SIMD registers, we need implementations that can offload even small computation units to accelerators, and we need parallelization strategies that are aware of modern hardware: Merely pinning processes to CPU hardware threads can have a 10-20% performance benefit today.

Finally, some of the most promising opportunities lie in the use of large-scale ensemble of thousands of loosely coupled simulations, where we have developed the COPERNICUS framework as a separate component that can be integrated with any simulation program. This makes it possible to use advanced sampling-based algorithms such as Markov State Modeling and Swarm-of-Strings as fully automated modules, which makes it possible to use some of the largest supercomputers in the world efficiently even for very small problems - and reach timescales that are orders of magnitude higher than what has been possible before.

James Phillips
Senior Research Programmer and NAMD Lead Developer,
University of Illinois at Urbana-Champaign

NAMD: Innovation Beyond Petascale

The highly parallel molecular dynamics code NAMD is used on the NCSA Blue Waters and ORNL Titan machines to perform petascale biomolecular simulations, including a 64-million-atom model of the HIV virus capsid published in 2013, based on scaling techniques presented at SC14.

Of increasing importance for enhancing sampling of more modestly sized systems are the variety of multiple-copy algorithms in NAMD, which are implemented as user-modifiable scripts driving inter-copy communication. Recent advances in NAMD 2.11 exploit the message-driven capabilities of the Charm++ parallel runtime system on which NAMD is based to support workflow-style programming, enabling the development of more adaptive sampling algorithms.

Looking forward, NAMD is being prepared for early science projects on the upcoming Intel and IBM/NVIDIA machines at Argonne and Oak Ridge. The talk will discuss what properties

of the NAMD design have enabled a smooth transition to modern supercomputers, and what changes are contemplated for the next generation.

Ross Walker
AMBER, San Diego Supercomputer Center, University of California San Diego

How do we map molecular dynamics simulations to future computing resources? Rethinking the problem

The path towards Exascale represents a new and challenging paradigm in high performance computing. Near-future HPC machines will combine complex interconnect hierarchies with substantial heterogeneity in the form of accelerators, non-uniform memory models and other technologies which will serve to make the programmer's job that much harder. Effectively utilizing such resources will be a significant challenge in the molecular biology arena and simply following the previous path of scaling up the size of the system being simulated will not be a viable solution.

One can certainly design a trillion atom system to simulate which could fill an ExaScale machine but the achievable simulation timescale is likely to be on the order of 6 to 7 orders of magnitude short of what is needed to converge the statistics for a system with 10^12 degrees of freedom. Instead, as a community, we need to look at ways to bring highly automated ensemble computing to to the field. Examples include ways to determine the free energy of drug binding of billions of compounds to a given target while simultaneously screening the binders against all other known targets. Such technology will be a boon to the drug discovery industry, providing not just a list of effective drug binders but also simultaneously checking for any undesired binding to unrelated targets which might lead to toxicity concerns. The challenges however, are immense. Such procedures will need to be completely automated from start to finish, errors will have to be detected and automatically handled and the resulting data will need to be analyzed on the fly since storing it for post process will simply be overwhelming.

The purpose of this talk will be to highlight some of the efforts that are being made in this field and provide a potential roadmap for discussion of the challenges that lie ahead.